

**Remarks/Arguments**

In response to the Final Office Action mailed August 13, 2004, Applicant submits this Amendment concurrently with a Request for Continued Examination. A complete listing of all pending claims is submitted herewith.

In the Office Action, claims 1-6 and 10-20 are finally rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,286,003 to Muta (the “Muta” patent). Claims 7-8 are finally rejected as being obvious over Muta in view of U.S. Patent No. 6,401,118 to Thomas (the “Thomas” patent). Claim 9 is finally rejected as being obvious over Muta in view of U.S. Patent No. 6,011,918 to Cohen et al. (the “Cohen” patent). By this amendment, claims 1 and 10 have been amended. The specification has been amended to change the priority claim, which was necessitated by the claims.

It is respectfully submitted that the claim amendments and accompanying remarks have been made to advance the claims to allowance, and as such, are completely responsive to the Final Office Action. No new matter has been added by the current amendments. For reasons to be set forth below, it is requested that the rejections be withdrawn and that the claims be allowed to issue.

**The Rejection under 35. U.S.C. §102**

Claims 1-6 and 10-20 are finally rejected under 35 U.S.C. § 102(e) as being anticipated by Muta. It is respectfully requested that this rejection be withdrawn.

Applicants’ arguments in the Amendment dated April 29, 2004 (‘the Amendment’) are preserved and repeated herein.

As previously argued in the Amendment, Muta describes a distributed system in which the function of generating graphics occurs at a slave server. More particularly, a “graphics engine 321” and a “display driver 325,” generate images at the “slave server 240” / “slave daemon 247” (Muta, Figure 8), e.g.:

“The drawing command monitor 323, which is located between the graphics engine 321 and the display driver 325, monitors all the APIs that

are called for *rewriting the GUI screen at the slave server 240*  
(Muta, column 11, lines 15-16, emphasis added);

“the same image as that *drawn on the GUI screen of the slave server 240*”  
(Muta, column 11, lines 55-56, emphasis added).

The drawing commands are then “stored in a drawing command buffer 327 in the slave server 240” (Muta, column 11, lines 17-19). Finally, *lower-level* drawing commands, such as rendering bit images, are transmitted to the master applet 215 (Muta, column 11, lines 19-28 and Figure 20). The above-described functionality of Muta is substantially identical to the frame buffer systems, such as Citrix and VNC, which were clearly distinguished from Applicants’ invention in the specification, as follows:

“[0010] A second approach to distributed computing involves creating a remote virtual frame buffer on the server, on which the application can draw, and then transporting the *resulting raster image* to the client....(emphasis added)

“[0011] The architecture of a remote frame buffer based application is illustrated in FIG. 2 for transmission between a server 24 and client 26. The application 28 is typically written using a standard user interface toolkit API 30, such as JAVA™ Foundation Class (hereinafter “JFC”),... and renders onto a remote virtual frame buffer 32. The resulting *pixel data 34* is transported across the network using a proprietary protocol, such as ICA ... to the client 26. The client viewer 36 receives the pixels and reconstructs the image, and then copies the image onto the client frame buffer 38 for presentation on the client’s display.” (emphasis added).

As discussed in Applicants’ specification, the approach of Muta and others has significant drawbacks, such as “demand[ing] high-bandwidth connections” because this approach “is essentially sending a video stream of computer-generated graphics from the server to the client.” (specification, ¶12.) Thus, the system of Muta functions poorly on slower connections: “When a remote frame buffer system is run on anything other than a high-speed LAN connection (e.g., 100 megabit per second over category 5 cabling), there is always a noticeable difference in position between the client (virtual) and server (real) mouse positions. On a slow modem link (e.g., 56.6 kilobit per second transmission over a standard telephone line), this makes highly interactive user interfaces difficult to control, and, in extreme cases, may even make the system unusable.” (specification, ¶14.) Moreover, the “compression” or “encoding” of Muta (Muta, column 11, lines 22-23)

“usually requires special hardware that can only handle one or two streams at a time, thereby eliminating the possibility of using the remote frame buffer approach on a current shared server. In addition, the use of lossy compression techniques introduces unwanted compression artifacts into the display, reducing the system’s usability, particularly when working with text and detailed graphics.” (specification, ¶13.)

Applicants’ invention, as recited in amended claim 1, overcomes these disadvantages of Muta and other techniques which render graphics on the server and then transmit a stream of low-level pixel image commands to the client for display on the client. In particular, amended claim 1 recites the step of “generating the message by the remote-capable component of the remote-capable user interface toolkit on the server in response to the invocation by the application, the message comprising a higher-level command to the user interface toolkit on the remote client to perform the function, such that the respective function is only performed at the remote client.”

Muta neither discloses nor suggests a remote-capable component which is configured to generate a message to the component on the remote client to perform the respective function on the remote client, such that the respective function is only performed on the remote client. As discussed above, Muta describes the function of drawing the image and generating pixel data performed on the slave daemon 247 (see, Muta, Figure 8, column 11, lines 12-21). In addition, since Muta performs the drawing function on the slaver server rather than on the client, Muta neither discloses nor suggests the step of generating a “message comprising a higher-level command to the user interface toolkit on the remote client to perform the function” (see, e.g., specification, ¶[0064] “higher-level semantics;” and ¶[0055] transmitting a command for JButton to render a complicated graphical item such as a button, or transmitting a command to install a proxy JFC event handler). Muta, in contrast, merely transmits low-level commands to render individual pixels on the client (see, Muta, Figure 20).

For the reasons discussed above, claim 1 is not believed anticipated by Muta. Claims 2-9 depend from claim 1 and are allowable at least for the reasons discussed above with respect to claim 1.

Claim 10 is allowable for the same reasons discussed above for claim 1. For example, claim 10 recites “a remote-capable user interface toolkit on the server by creating a remote-capable component which is configured to interact with the application according to the application programming interface and which is configured to generate a message, the message comprising a higher-level command to the component on the remote client to perform the respective function on the remote client, such that the respective function is only performed on the remote client.” Muta neither discloses nor suggests such a feature. Accordingly, Claim 10 is not believed anticipated by Muta. Claims 11-20 depend from claim 10 are allowable at least for the reasons discussed above for claim 10. It is therefore requested that the rejection of claims 1-6 and 10-20 under 35 U.S.C. 102(e) be withdrawn.

### **The Rejection under 35. U.S.C. §103**

Claims 7-8 were rejected as being obvious under 35. U.S.C. §103 over Muta in view of Thomas. Applicants’ arguments in the Amendment are preserved herein. Claims 7-8 depend from claim 1 and are believed allowable at least for the reasons discussed above with respect to claim 1.

Further, Applicants believe that the combination of Muta and Thomas is an improper “obvious to try” rejection. In *In re Eli Lilly & Co.*, 902 F.2d 943 (Fed.Cir.1990), the Federal Circuit explained that [a]n “obvious-to-try” situation exists when a general disclosure may pique the scientist's curiosity, such that further investigation might be done as a result of the disclosure, but the disclosure itself does not contain a sufficient teaching of how to obtain the desired result, or that the claimed result would be obtained if certain directions were pursued.”

In this case, the Examiner refers to a *general* interest “to improve throughput and availability of an application.” (Final Office Action, page 12.) However, section 103 requires “a showing that an artisan of ordinary skill in the art at the time of invention, confronted by the same problems as the inventor and with no knowledge of the claimed invention, would select the various elements from the prior art and **combine them in the claimed manner.**” *Ruiz v. A.B. Chance Co.*, 357 F.3d 1270 (Fed.Cir.2004) (emphasis

supplied.) To support a rejection for obviousness, the suggestion to combine must be “clear and particular.” *In re Dembiczak*, 175 F.3d 994, 999 (Fed. Cir. 1999).

In this case, the Examiner has not specifically identified how Muta in view of Thomas would direct one to “select the various elements from the prior art and combine them in the claimed manner” as recited in claim 7. The purported advantage of Thomas appears in the specification as follows:

“its back-end (search engine) and front-end (user interface) are designed to operate independently of each other, thus allowing greater throughput and availability of the system as a whole.” (Thomas, column 2, lines 61-65).

This suggestion or teaching does not provide motivation to make the *particular modification* of Muta to include “identifying information from the database in response to the user-defined request” and “asynchronously transmitting a command to the remote client to render the information from the database,” nor does the Examiner assert that any teaching or motivation exists to suggest that such modification would actually result in improvements in throughput and availability. Moreover, Muta operates in a different manner than recited in claim 7, since Muta performs the function on the slave server rather than the client. It is respectfully submitted that there is insufficient motivation to make such modification. Therefore claim 7, and claim 8, which depends from claim 7, are not obvious and such rejection of claims 7-8 under 35 U.S.C. §103 should be withdrawn.

Claim 9 was rejected as being obvious over Muta in view of U.S. Patent No. 6,011,918 to Cohen et al. (the “Cohen” patent). Applicants’ arguments in the Amendment dated April 29, 2004 (‘the Amendment’) are preserved herein. Claim 9 depends from claims 1 and is believed allowable at least for the reasons discussed above with respect to claims 1.

The Examiner makes a *general* statement that it is obvious to supplement a distributed sever and remote client of Muta with code generating features of Cohen. As discussed above with respect to claims 7-8 above, the teachings of Muta in view of

Cohen do not provide sufficient motivation to “select the various elements from the prior art and combine them in the claimed manner” as recited in claim 9.


To establish obviousness, the prior art must teach or suggest all of the claim limitations. *In re Royka*, 490 F.2d 981 (CCPA 1974) Neither Muta nor Cohen, either alone or in combination, disclose or suggest the step of “substituting the portion of the code relevant to executing the function with the portion of code configured to issue the remote command to execute the function” as recited in claim 9. The Examiner acknowledges that Muta neither discloses nor suggests this step. The Examiner attempts to make up for the deficiency of Muta by stating that Cohen generally discloses “generating code automatically.” In making this rejection, the Examiner disregards the actual claim language, as recited above. To support a rejection under 35. U.S.C. §103, “[a]ll words in a claim must be considered in judging the patentability of that claim against the prior art.” *In re Wilson*, 424 F.2d 1382, 1385 (CCPA 1970). Accordingly, since neither reference discloses or suggests “substituting the portion of the code relevant to executing the function with the portion of code configured to issue the remote command to execute the function,” nor has the Examiner asserted that either reference teaches such subject matter, claim 9 is believed non-obvious over the combined references. It is requested that the rejection be withdrawn.

**Conclusion**

In view of the foregoing, Claims 1-20 are believed allowable, and this application is believed to be in condition for formal allowance. Prompt and favorable action is respectfully requested.

Respectfully submitted,

Dated: December 15, 2004

By   
Walter M. Egbert, III  
Patent Office Reg. No. 37,317  
Attorney for Applicants  
BAKER BOTTS, L.L.P.  
30 Rockefeller Plaza  
New York, NY 10112  
(212) 408-2561